REMARKS

This Amendment is responsive to the Office Action mailed March 3, 2009, and is filed concurrently with the fees for a one-month extension of time (large Entity).

Claims 1-16, 82, 84-90 are withdrawn.

Claims 17-21, 24 and 25 were rejected as being unpatentable over a four-way combination of Gunyakti, Yip, Fieres and Lambert. Reconsideration and withdrawal of these rejections are respectfully requested, for the reasons to follow.

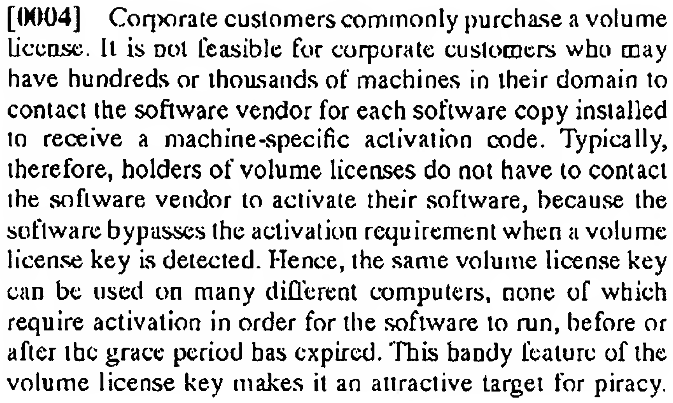
Independent claim 17

Claim 17, as amended, recites:

> **producing a separate and unique PKI certificate for each of the plurality of executable software components subject to receiving certification within each gaming machine, each software component subject to receiving certification including a unique identifier;**

Claim 17, therefore, requires a separate and unique PKI certificate for each of the plurality of executable software components subject to receiving certification. The Office relies on **Gunyakti** for this feature and cites Fig. 2 and paragraphs 0026-0028 in support thereof.

However, the very purpose of Gunyakti is to service volume licenses (identified by Volume License Keys or VLKs), and specifically not individual licenses for each piece of software. This is because a large company deploying potentially thousands of copies of a same piece of software does not want, according to Gunyakti, to manage corresponding thousands of licenses:



> [0004] Corporate customers commonly purchase a volume license. It is not feasible for corporate customers who may have hundreds or thousands of machines in their domain to contact the software vendor for each software copy installed to receive a machine-specific activation code. Typically, therefore, holders of volume licenses do not have to contact the software vendor to activate their software, because the software bypasses the activation requirement when a volume license key is detected. Hence, the same volume license key can be used on many different computers, none of which require activation in order for the software to run, before or after the grace period has expired. This handy feature of the volume license key makes it an attractive target for piracy.

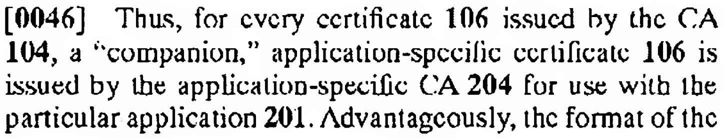
Paragraph 0026 details the general client/server/vendor/network architecture of their proposed system. Paragraph 0027 explains that the volume license key (VLK) is embedded in a large file (greater than 1.44 MB) and may be provided on a CD-ROM. Paragraph 0028 explains how an activation code on client machines reads the contents of the volume license to make sure that the volume license has not been tampered with.

Neither these paragraphs, nor Gunyakti as a whole, teach or suggest producing a separate and unique license for each of the plurality of executable software components, as required by the claim. In fact, it is the exact contrary: Gunyakti teaches volume licenses that cover potentially thousands of copies of a piece of software.

Next, the Office relies upon **Yip** for the same teaching of producing a separate and unique PKI certificate for each of the plurality of executable software components subject to receiving certification within each gaming machine, and points to Figs. 2 and 3 and paragraphs 0048 and 0046.

In Yip, a conventional Certificate Authority (CA) issues a certificate 106 and an application-specific CA issues a corresponding application-specific certificate 206. See paragraph 0042. The certificate 106 and application certificate 206 are linked, such that when the certificate 106 is revoked, the application-specific certificates are also preferably revoked. See paragraph 0044.

.Thus, the application-specific certificate 206 is a "companion" to the certificate 106 (note error wherein second instance of "106" in the passage below should be "206"):



> [0046] Thus, for every certificate **106** issued by the CA **104**, a "companion," application-specific certificate **106** is issued by the application-specific CA **204** for use with the particular application **201**. Advantageously, the format of the

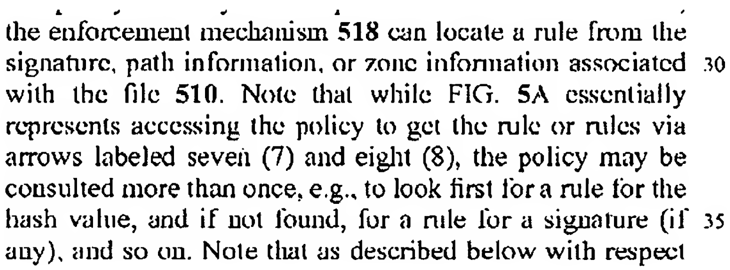
Note, however, that claim 17 recites:

> **code signing each executable software component subject to receiving certification with its respective separate and unique PKI certificate, each respective PKI certificate being uniquely identified at least by a unique identifier that is uniquely associated with the executable software component** ***such that identical executable software components in different ones of the plurality of gaming machines of the network connected gaming system are associated with identical identifiers and are code signed with identical PKI certificates, such that non-identical executable software components in different ones of the plurality of gaming machines are associated with separate and different identifiers and are code signed with separate and different PKI certificates and such that no two non-identical executable software components in different gaming machines are code signed with a same PKI certificate,*** **... (italics for emphasis)**

As the application-specific certificate 206 is the companion of certificate 106 and is granted/revoked concurrently, identical executable software components in different ones of the plurality of gaming machines, in Yip, would be associated with different PKI certificates, as each gaming machine, in Yip, would receive a different certificate 106 and corresponding different application-specific certificates 206. There is no teaching or suggestion in Yip otherwise. That is, there is no teaching or suggestion in Yip of providing identical application-specific certificates in different machines for identical executable software components, as the conventional certificates 106 (to which the application-specific certificates are associated) would be different for each user/machine. In other words, the CA in Yip would not issue identical certificates 106 to more than one user/user nor would the CA issue identical companion application-specific certificates 206 to more than on user/machine, as each certificate 106 is different and as the application-specific certificates 206 are companions to such different certificates 106.

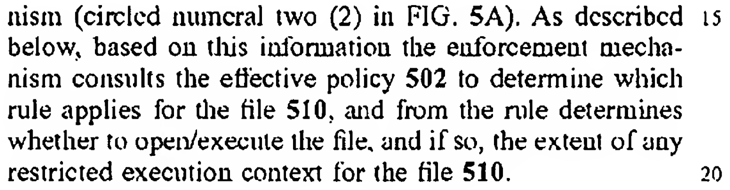
Turning now to **Fieres**, the undersigned notes the following. Fieres issues application certifications to insure that applications operate at the proper cryptographic level granted for that application by an application domain authority 22. However, there is no teaching or suggestion in Fieres that "identical executable software components in different ones of the plurality of gaming machines of the network connected gaming system are associated with identical identifiers and are code signed with identical PKI certificates", as claimed. Nor is there any teaching or suggestion in Fieres that "non-identical executable software components in different ones of the plurality of gaming machines are associated with separate and different identifiers and are code signed with separate and different PKI certificates", as claimed. Lastly, Fieres does not teach or suggest that "no two non-identical executable software components in different gaming machines are code signed with a same PKI certificate", as claimed herein. In fact, it is highly unlikely that, in the context of the distribution of cryptographic capabilities, that Fieres would allow identical executable components in different machines to have identical certificates, as required herein. Such would surely defeat the security measures.

Lambert was relied on for an alleged teaching of a software restriction policy certificate rule for each of the plurality of executable software components, as set forth at the bottom of page 4 of the outstanding Office Action. However, Lambert does not teach a software restriction policy certificate rule for each executable software component. Quite to the contrary, Lambert teaches one rule for an entire security level for executing executable software (see Abstract, lines 3-4). Lambert also teaches a hierarchy of rules, to help distinguish which rule to use should a piece of software having multiple classifications (see Abstract, last sentence). In Column 15, Lambert teaches how rules are selected...



the enforcement mechanism **518** can locate a rule from the signature, path information, or zone information associated with the file **510**. Note that while FIG. 5A essentially represents accessing the policy to get the rule or rules via arrows labeled seven (7) and eight (8), the policy may be consulted more than once, e.g., to look first for a rule for the hash value, and if not found, for a rule for a signature (if any), and so on. Note that as described below with respect

...and how rules determine the execution of the file.



nism (circled numeral two (2) in FIG. 5A). As described below, based on this information the enforcement mechanism consults the effective policy **502** to determine which rule applies for the file **510**, and from the rule determines whether to open/execute the file, and if so, the extent of any restricted execution context for the file **510**.

In Lambert, therefore, there is no on-to-one relationship (a SRP for each executable software components) with executable software components and rules, as required by claim 17:

> **configuring a software restriction policy certificate rule for each of the plurality of executable software components and enforcing each of the software restriction policy certificate rules to allow execution of only those executable software components whose code signed PKI certificate is determined to be authorized.**

<u>Considering now the applied references in combination, as is proper under 35 USC §103</u>, the undersigned notes that application-specific certificates exist, as taught by Fieres and Yip. However, the applied combination still fails to teach or to suggest:

> **producing a separate and unique PKI certificate for each of the plurality of executable software components subject to receiving certification within each gaming machine, each software component subject to receiving certification including a unique identifier;**

or

> **code signing each executable software component subject to receiving certification with its respective separate and unique PKI certificate, each respective PKI certificate being uniquely identified at least by a unique identifier that is uniquely associated with the executable software component** ***such that identical executable software components in different ones of the plurality of gaming machines of the network connected gaming system are associated with identical identifiers and are code signed with identical PKI certificates, such that non-identical executable software components in different ones of the plurality of gaming machines are associated with separate and different identifiers and are code signed with separate and different PKI certificates and such that no two non-identical executable software components in different gaming machines are code signed with a same PKI certificate,*** **... (italics for emphasis)**

or

> **configuring a software restriction policy certificate rule for each of the plurality of executable software components and enforcing each of the software restriction policy certificate rules to allow execution of only those executable software components whose code signed PKI certificate is determined to be authorized.**

... as claimed in independent claim 17. In fact, the Office's primary reference to Gunyakti teaches volume licenses (the antithesis of the claimed embodiment) and Yip teaches "companion" certificates tied to (and revoked along with) a conventional certificate. Moreover, Lambert teaches a one-to-many relationship between the rules and the applications, such that the correct rule must be determined before execution of the application is allowed. Not only are the claimed elements absent from the applied combination or any of the individual references, but the combination as a whole appears to teach away from an embodiment that include separate and unique PKI certificates, the code signing step or configuring SRPs for each executable software component, as Gunyakti teaches the exact opposite, as Yip teaches that the conventional and application-specific certificates are linked together (one is the "companion" to the other) and as Lambert teaches a one-to-many relationship between the security rules and the applications.

Indeed, there is no teaching or suggestion in the applied references that would lead a person of ordinary skill in the art to develop the embodiment of claim 17, which requires separate and unique PKI certificates for each executable software component, code signing each executable software component with its respective individual and separate certificate or configuring a software restriction policy certificate rule for each of the plurality of executable software components and enforcing each of the software restriction policy certificate rules to allow execution of only those executable software components whose code signed PKI certificate is determined to be authorized, as claimed herein.

Independent claim 20

Claim 20 recites:

> code signing each authorized software component with a PKI certificate such that identical authorized software components in different ones of the constituent computers are code signed with identical PKI certificates, such that non-identical authorized software components in different ones of the constituent computers are code signed with separate and different PKI certificates and such that no two non-identical authorized software components in different ones of the constituent gaming machines are code signed with a same PKI certificate;
>
> configuring a separate software restriction policy for each authorized software component in each of the constituent computers of the gaming system, and associating the configured separate software restriction policy with the PKI certificate with which the authorized software component was code signed;
>
> enforcing the associated software restriction policy for each code signed authorized software component such that each code signed authorized software component in each of the constituent computers of the gaming system must be authorized to run by its associated separate software restriction policy.

The arguments presented above relative to claim 17 are equally applicable to claim 20. Rather than repeat these here, reference is made to the arguments above, which are incorporated herein in their entirety, as if repeated here in full.

Independent claim 22

Claim 22 was rejected as being unpatentable over Lambert-Gunyakti-Yip.

Claim 22 recites, similarly to claims 17 and 20:

> configuring a separate and unique certificate software restriction policy for each authorized executable software component of each of the constituent computers of the gaming system such that the each authorized executable software component in each of the constituent computers of the gaming system must be authorized to run by its associated separate software restriction policy;
>
> code signing each authorized software component with a PKI certificate such that identical authorized software components in different ones of the constituent computers are code signed with identical PKI certificates, such that non-identical authorized software components in different ones of the constituent computers are code signed with separate and different PKI certificates and such that no two non-identical authorized

> **software components in different ones of the constituent gaming machines are code signed with a same PKI certificate;**

Although Lambert does teach rules based upon a path in Column 13, the applied combination of Lambert-Gunyakti-Yip does not teach or suggest the configuring and code signing steps recited above, nor, by extension, the claimed step of:

> **enforcing the certificate software restriction policy configured for each of the code signed authorized executable software components of each of the constituent computers of the gaming system, and**

for the same arguments as were presented above relative to claim 17. These same arguments are incorporated herein in their entirety, as if repeated here in full.

Independent claim 24

Claim 24 recites:

> **producing a separate and unique PKI certificate for each of the plurality of executable software components within the gaming system subject to receive certification, each respective PKI certificate being associated with a unique identifier that is uniquely associated with the executable software component such that identical executable software components in different ones of the plurality of gaming machines of the network connected gaming system are associated with identical identifiers and are code signed with identical PKI certificates, such that non-identical executable software components in different ones of the plurality of gaming machines are code signed with separate and different PKI certificates and such that no two non-identical executable software components in different gaming machines are code signed with a same PKI certificate;**

... for which the arguments above are applicable.

Moreover, claim 24 also recites:

> **code signing each software component subject to receive certification with its respective separate and unique PKI certificate;**
>
> **configuring a certificate software restriction policy for each of the respective separate and unique PKI certificates, and**
>
> **enforcing the certificate software restriction policy for each of the respective separate and unique PKI certificates.**

In contradistinction, the primary reference to Gunyakti advocates volume licenses, Yip advocates companion application-specific certificates and Lambert calls for a hierarchy of rules to enable the application of a specific rule to a specific application. The applied combination does not teach code signing each software component subject to receive certification with its respective separate and unique PKI certificate (compare to Gunyakti's volume licenses), configuring a certificate software restriction policy for each of the respective separate and unique PKI certificates (compare with the one-to-many relationship of Lambert's rules to the applications) or enforcing the certificate software restriction policy for each of the respective separate and unique PKI certificates, as claimed herein.

Independent claim 25

Independent claim 25, similarly to the claims above, recites:

> **for each of the plurality of gaming machines of the network connected gaming system:**
>
> **code signing each authorized executable software component with a separate PKI certificate that is unique to the authorized software component such that identical executable software components in different ones of the plurality of gaming machines of the network connected gaming system are code signed with identical PKI certificates, such that non-identical authorized software components in different ones of the plurality of gaming machines are code signed with separate and different PKI certificates and such that no two non-identical authorized software components in different gaming machines are code signed with a same PKI certificate;**

... and the arguments advanced hereinabove are incorporated by reference.

However, claim 25 also recites:

> **packaging the code signed authorized software components into an installation package;**
>
> **configuring install policies to install each code signed authorized executable software component contained in the installation package;**
>
> **configuring certificate rule policies to allow execution of the installed code signed authorized executable software component;**
>
> **configuring enforcement of the policies.**

Although claim 25 includes additional new recitations, the Office limited its examination of this claim to a statement that it "encompasses limitations that are similar to claim 17" and rejected the claim on the same rationale.

However, as the Office will note, claim 17 also includes a recitation of "packaging the code signed authorized software into an installation package", which finds no counterpart in any of the previous independent claims and no counterpart in any of the applied references, whether considered singly or in combination. Similarly, the claim also calls for configuring install policies and for configuring enforcement of the policies. The applied combination does not teach or suggest any such embodiment, nor has the office pointed to any such teachings in the applied combination.

As the claims define embodiments that are not taught or suggested in the applied four-way combination of references, reconsideration and withdrawal of the 35 USC 103(a) rejections are believed to be warranted. The same, therefore, is respectfully requested.

Applicants' attorney believes that the present application is now in condition for allowance and passage to issue. If any unresolved issues remain, the Examiner is respectfully invited to contact the undersigned attorney of record at the telephone number indicated below, and whatever is required will be done at once.

Respectfully submitted,

Date: June 17, 2009

By: Alan W. Young
Alan W. Young
Attorney for Applicants
Registration No. 37,970

YOUNG LAW FIRM, P.C.
4370 Alpine Rd., Ste. 106
Portola Valley, CA 94028
Tel.: (650) 851-7210
Fax: (650) 851-7232

\\Ylfserver\ylf\CLIENTS\JMG\5858 (Trusted Game Download)\5858 AMEND 5.doc